Ministerstwo Nauki i Szkolnictwa Wyższego

Organizacja IX Konferencji Modelowanie Matematyczne w Fizyce i Technice (MMFT 2017) - zadanie finansowane w ramach umowy 829/P-DUN/2017 ze środków Ministra Nauki i Szkolnictwa Wyższego przeznaczonych na działalność upowszechniającą naukę.

LINEAR RECURRENCES ALGORITHM FOR SOLVING TRIDIAGONAL SYSTEMS WITH IMPLEMENTATION IN MAPLE

Lena Łacińska, Institute of Mathematics, Czestochowa University of Technology

Introduction

The subject of consideration is a linear algebraic tridiagonal system for n unknowns represented by a matrix equation

$$\mathbf{A}_n \cdot \mathbf{x} = \mathbf{d} \qquad (1)$$

where

$$\mathbf{A}_n = \begin{bmatrix} a_1 & c_1 & 0 & \dots & \dots & 0 \\ b_2 & a_2 & c_2 & \ddots & & \vdots \\ 0 & b_3 & a_3 & c_3 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & & \ddots & b_{n-1} & a_{n-1} & c_{n-1} \\ 0 & \dots & \dots & 0 & b_n & a_n \end{bmatrix},$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} d_1 \\ d_2 \\ d_3 \\ \vdots \\ d_{n-1} \\ d_n \end{bmatrix}$$

We are to use Maple system to implement an algorithm which is based on results presented in [1].

Linear recurrences algorithm

Bearing in mind the considerations presented in [1] we conclude that solution to (1) can be obtained in 3 steps.

Step 1. Calculation of the determinant W_n of the matrix $\mathbf{A}_n$

$$\begin{cases} W_1 = a_1,\ W_2 = a_1 a_2 - b_2 c_1, \\ W_n = a_n W_{n-1} - b_n c_{n-1} W_{n-2},\ n > 2 \end{cases}$$

Step 2. Calculation of $W_n^{x_1}$ which is the determinant of the matrix obtained from matrix $\mathbf{A}_n$ by replacing its first column by the vector $\mathbf{d}$.

In order to obtain determinant $W_n^{x_1}$ we must take into account the second order nonhomogeneous linear recurence equation

$$W_n^{x_1} = a_n W_{n-1}^{x_1} - c_{n-1} b_n W_{n-2}^{x_1} - (-1)^n d_n \prod_{i=1}^{n-1} c_i, \quad n > 2$$

together with initial conditions

$$W_1^{x_1} = d_1,\ W_2^{x_1} = d_1 a_2 - d_2 c_1$$

Step 3. Solution to the linear algebraic tridiagonal system (1). Bearing in mind [1] we conclude that this problem comes down to resolving linear recurrence equation

$$x_k = \frac{1}{c_{k-1}}\left(d_{k-1} - b_{k-1} x_{k-2} - a_{k-1} x_{k-1}\right)$$

with initial conditions

$$x_1 = \frac{W_n^{x_1}}{W_n}, \quad x_2 = \frac{1}{c_1}(d_1 - a_1 x_1)$$

Implementation in Maple

To this end let us consider the tridiagonal linear system of algebraic equation which has 2-Toeplitz structure and consists of 100 unknowns with main matrix of the form

$$\mathbf{A}_{100} = \begin{bmatrix} 1 & -1 & 0 & \cdots & \cdots & \cdots & 0 \\ 1 & 2 & 2 & \ddots & & & \vdots \\ 0 & 3 & 1 & -1 & \ddots & & \vdots \\ \vdots & \ddots & 1 & 2 & 2 & \ddots & \vdots \\ \vdots & & \ddots & 3 & 1 & \ddots & 0 \\ \vdots & & & \ddots & \ddots & \ddots & -1 \\ 0 & \cdots & \cdots & \cdots & 0 & 1 & 2 \end{bmatrix}$$

and the vector of right-hand-sides of the equations has the form

$$\mathbf{d} = [d_i]_{1\times 100},\ d_i = i+1,\ i = 1,2,..,100$$

In order to solve this system of equations using the above presented recurrence method we implement the proper syntax to Maple system, [2].

We start with declaration of all data

```
n := 100
a := Array([seq(0, i = 1..n)])
for i from 1 to n do
 if i mod 2 = 1 then
   a[i] := 1 :
 else
   a[i] := 2 :
 end if
end do :

b := Array([seq(0, i = 1..n)])
for i from 2 to n do
 if i mod 2 = 1 then
   b[i] := 3 :
 else
   b[i] := 2 :
 end if
end do :

c := Array([seq(0, i = 1..n)])
for i from 1 to n-1 do
 if i mod 2 = 1 then
   c[i] := -1 :
 else
   c[i] := 2 :
 end if
end do :

d := Array([seq(i+1, i = 1..n)]) :
```

Subsequently we implement steps 1-3

Step 1.

```
W := Array([seq(0, i = 1..n)])
W[1] := a[1] :
W[2] := a[1]·a[2] - b[2]·c[1] :
for i from 3 to n do
   W[i] := a[i]·W[i-1] - b[i]·c[i-1]·W[i-2] :
end do :
```

Step 2.

```
W1 := Array([seq(0, i = 1..n)])
W1[1] := d[1] :
W1[2] := d[1]·a[2] - d[2]·c[1] :
for i from 3 to n do
W1[i] := a[i]·W1[i-1] - b[i]·c[i-1]·W1[i-2] - (-1)^i ·
         ·d[i]·mul(c[k], k = 1..i-1) :
end do :
```

Step 3.

```
x := Array([seq(0, i = 1..n)])
x[1] := W1[n] / W[n] :
x[2] := 1/c[1] · (d[1] - a[1]·x[1]) :
for i from 3 to n do
x[i] := 1/c[i-1] · (d[i-1] - b[i-1])·x[i-2] +
      - a[i-1]·x[i-1] :
end do :
```

Conclusions

It has to be emphasized that the presented algorithm can be used without necessity to impose any conditons on elements of main matrix of the analyzed linear system of equations.

References

[1] Borowska J., Łacińska L., Application of second order inhomogeneous linear recurrences to solving a tridiagonal system, Journal of Applied Mathematics and Computational Mechanics, 2016, 15(2), 5-10

[2] Adams P., Smith K., Vyborny R., Introduction to Mathematics with Maple, World Scientific 2004.